

Definição da Arquitetura Smart Garden

Grupo:

GABRIEL FORMIGONI DOS SANTOS NETO
MARCOS VINÍCIUS DE MORAES
PHELIPE GABRIEL DE SOUZA CECILIO

Links:

github: <https://github.com/Markoro-Original/SmartGarden>

Introdução

Este documento descreve a arquitetura proposta para o sistema Smart Garden, assim como os elementos e visões que levaram à decisão arquitetural.

A arquitetura escolhida pela equipe foi a de **Microserviços** para modularidade, escalabilidade e independência dos componentes, além de integrar um Gêmeo Digital Preditivo para previsões avançadas com base em padrões históricos.

Principais Componentes

- **Sensores**

Os sensores são responsáveis pela coleta de dados ambientais e estão conectados ao Esp32 para processamento local inicial, se necessário.

- **Esp32**

O Esp32 é encarregado de processar dados localmente e de estabelecer comunicação bidirecional com o Broker MQTT para troca eficiente de mensagens com outros componentes do sistema.

- **Broker MQTT**

O Broker MQTT facilita a comunicação assíncrona entre o Esp32, o servidor e a interface web e mobile, por onde o usuário irá interagir com o sistema.

- **Servidor**

O servidor consiste de microsserviços integrados via API REST, que trabalham para processar, analisar e registrar dados coletados, além de executar funcionalidades específicas de cadastro e login.

- **Frontend**

O frontend é a interface do usuário que permite a interação com o sistema. A priori, o frontend será implementado em versão web e mobile.

- **Banco de dados**

O banco de dados é acessado pelo servidor e compartilhado entre os microsserviços. O banco de dados será utilizado para guardar dados de login de usuários, e dados lidos pelos sensores, para posteriormente serem analisados e utilizados no gêmeo digital.

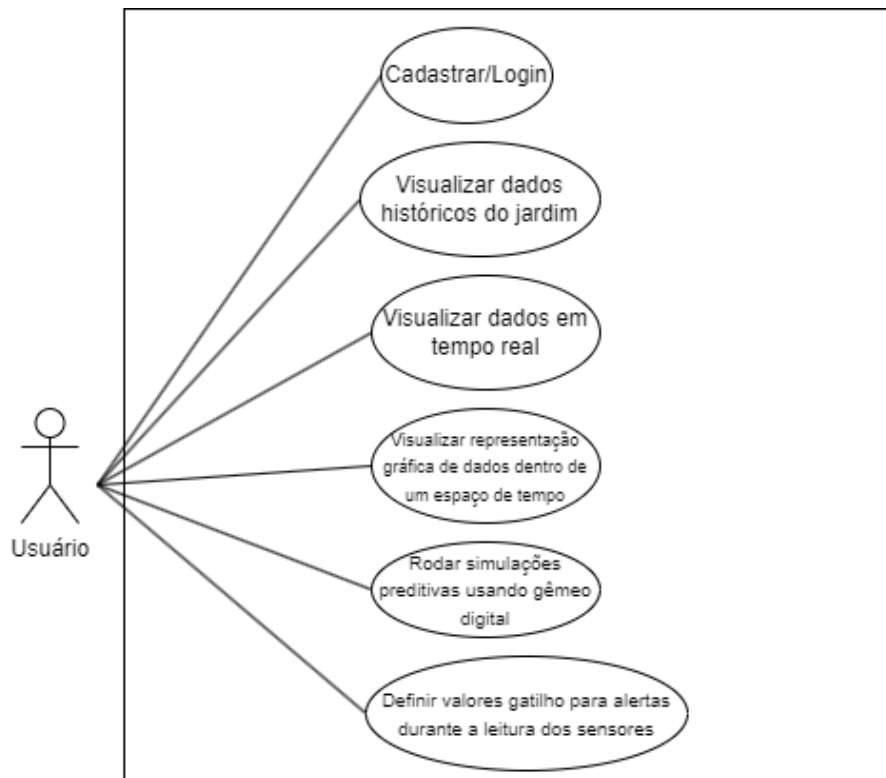
- **Gêmeo Digital Preditivo**

O Gêmeo Digital Preditivo utiliza os dados coletados pelos sensores em tempo real, assim como aqueles registrados em banco e, utiliza algoritmos de aprendizado de máquina para criar um modelo preditivo, permitindo previsões avançadas com base em padrões históricos.

Visões

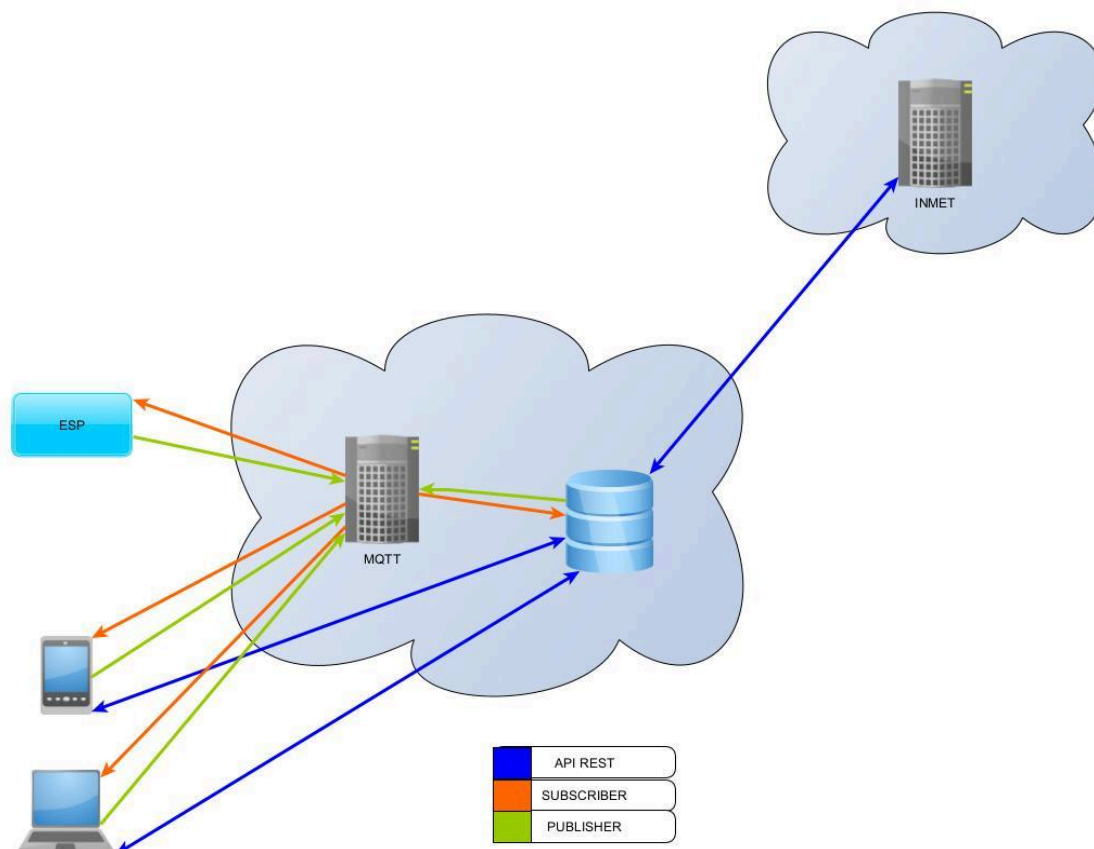
Visão de Casos de Uso

Os casos de uso apresentados no documento de proposta do projeto foram usados para gerar um diagrama de casos de uso representando as principais interações entre usuário e sistema, além de destacar funcionalidades suportadas pela arquitetura escolhida.



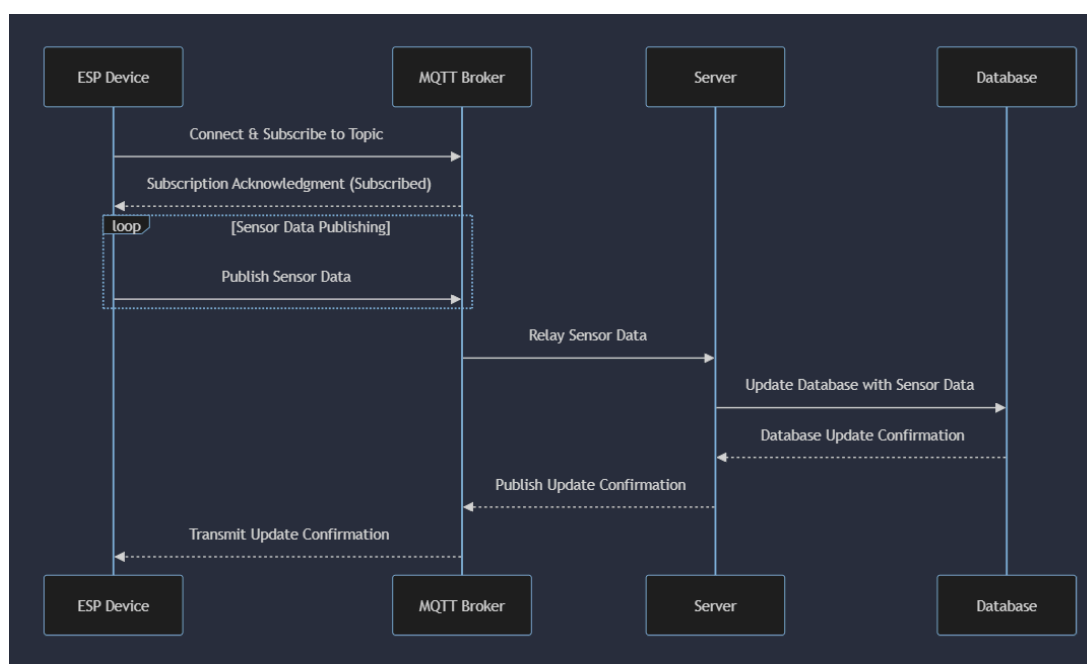
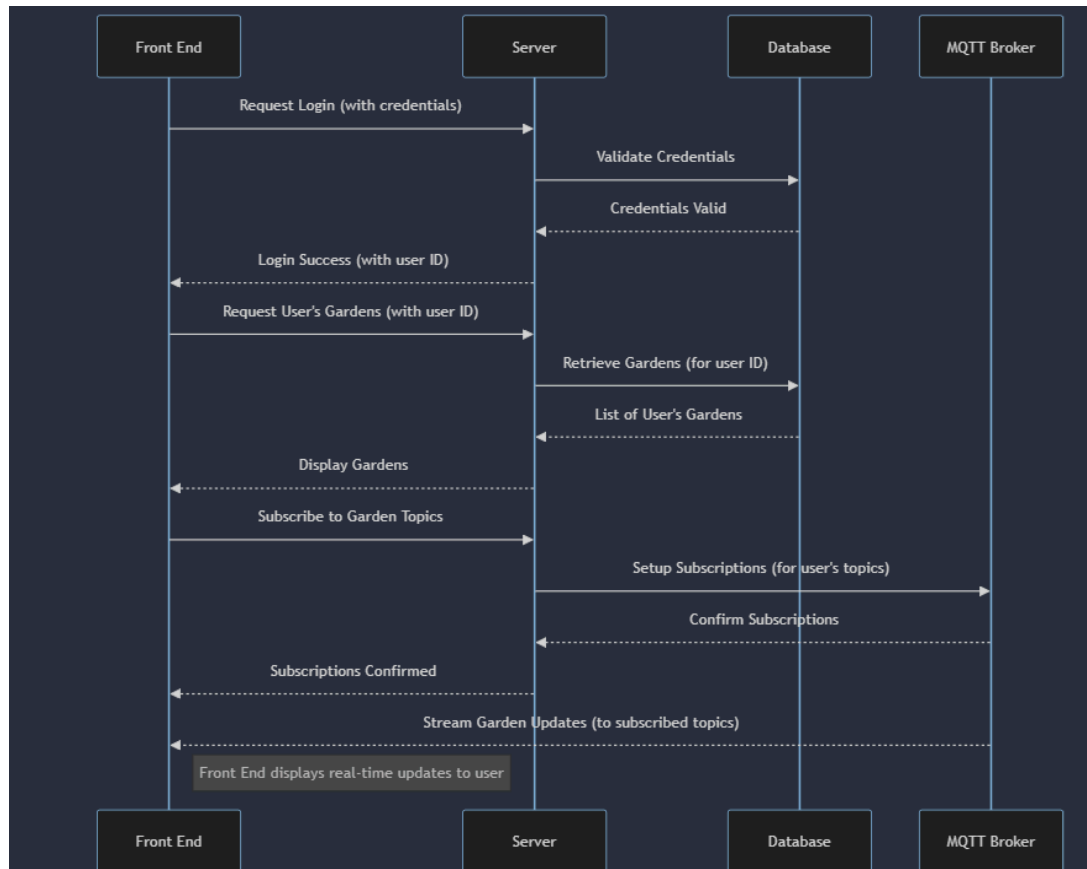
Visão de Interação

A visão de interação destaca como os diversos componentes interagem para realizar os casos de uso identificados.



Visão de Processos

A visão de processo fornece uma representação das atividades e interações entre os diferentes componentes ao longo do uso do sistema. Para destacar tais processos e interações, foram utilizados diagramas de fluxo do uso do sistema.



Microsserviços

Com base nos componentes destacados e nas visões geradas, a equipe optou por uma arquitetura de microsserviços incorporando princípios de modularidade, escalabilidade e independência das partes do sistema, e a interação entre os componentes é realizada por meio de API REST e via mensageria Broker MQTT.

Nesse estilo arquitetural, a aplicação é dividida em serviços pequenos, independentes e escaláveis. Cada serviço é projetado para executar uma função específica e pode ser implantado, atualizado e escalado independentemente dos outros. O que torna essa escolha ideal para o sistema Smart Garden, que conta com tantos componentes distintos, mas com uma grande interação entre eles.

A arquitetura de microsserviços oferece uma abordagem modular e escalável ideal para sistemas IoT, sendo projetada para escalar horizontalmente, permitindo adição fácil de recursos e serviços para lidar com aumentos na carga do sistema. Além de permitir a incorporação de segurança em cada camada da arquitetura, incluindo comunicação segura via TLS/SSL, gerenciamento de autenticação e autorização, e implementação de práticas recomendadas de segurança em microsserviços.

Microsserviços que serão utilizados:

- **Serviço de Análise de Dados**

Recebe dados brutos dos sensores e realiza análises e transformações para depois armazenar os dados processados em um banco de dados ou utilizá-los em tempo real no Gêmeo Digital.

- **Serviço de Notificação**

Monitora os dados recebidos pelo Esp 32, assim como o Gêmeo Digital Preditivo e gera alertas ou notificações com base em eventos significativos ou desvios previstos.

- **Serviço de Autenticação e Autorização**

Gerencia a autenticação de usuários e dispositivos, controlando o acesso aos serviços e dados do sistema.

- **API Gateway**

Fornece uma interface unificada, roteando solicitações para os microsserviços apropriados, gerenciando autenticação, autorização e outras funcionalidades de gateway.

- **Serviço de Configuração**

Gerencia configurações globais do sistema, permitindo atualizações sem a necessidade de reiniciar toda a aplicação.

Atributos de Qualidade

Escalabilidade:

O sistema deve ser capaz de lidar com a adição de novos sensores ou funcionalidades sem comprometer seu desempenho.

A arquitetura do sistema deve ser projetada para expansão, considerando um aumento no número de usuários ou dispositivos.

Métricas:

Tempo de Resposta: Mede o tempo que o sistema leva para responder a uma solicitação do usuário.

Taxa de Transmissão de Dados: Avalia a eficiência na transmissão de dados entre dispositivos, especialmente relevante em sistemas IoT.

Número de usuários simultâneos: Mede quantos usuários o sistema pode suportar simultaneamente sem degradação significativa do desempenho.

Disponibilidade:

O sistema deve estar disponível para coletar dados e fornecer informações aos usuários continuamente.

O sistema deve garantir uma infraestrutura de comunicação estável para a transmissão de dados do microcontrolador para o servidor Broker MQTT.

Métricas:

Uptime: Representa o período durante o qual um sistema está operacional e disponível para os usuários.

Downtime: Representa o período durante o qual o sistema está indisponível ou inoperante.

Confiabilidade:

Os sensores devem ser precisos e confiáveis na coleta de dados.

O gêmeo digital deve refletir com precisão o estado atual do jardim com base nos dados coletados.

Métricas:

Tempo Médio entre falhas (MTBF): Mede o tempo médio entre as ocorrências de falhas no sistema.

Tempo Médio de recuperação (MTTR): Avalia o tempo médio necessário para restaurar o sistema após uma falha.

Segurança:

O sistema deve implementar práticas de segurança para proteger os dados coletados, transmitidos e armazenados.

Além disso, devem ser utilizados mecanismos de autenticação e autorização para garantir que apenas usuários autorizados acessem as informações do jardim.

Métricas:

Taxa de Falhas de Autenticação: Mede a frequência com que ocorrem falhas no processo de autenticação.

Tempo de Detecção de Intrusões: Avalia quanto tempo leva para o sistema detectar e responder a atividades suspeitas.

Interoperabilidade:

Garantir a integração eficiente com diferentes tipos de sensores e dispositivos IoT.

Facilitar a expansão do sistema para suportar novos padrões e tecnologias emergentes.

Métricas:

Testes de Integração: Mede a eficácia dos testes de integração realizados para garantir que diferentes componentes e sistemas possam se integrar sem problemas.